

14
Login remoto

Você é o responsável pela criação de um mecanismo de login remoto de uma aplicação Xamarin Forms da Alura Car.

Seguindo a especificação, você decide montar a requisição necessária para fazer o login utilizando a extensão Postman do Google Chrome:

POST https://aluracar.herokuapp.com/login

Headers (1)

Body (x-www-form-urlencoded)

Body

```

1 [ ]
2   "usuario": {
3     "id": 1,
4     "nome": "João da Silva",
5     "dataNascimento": "30/01/1990",
6     "telefone": "11998877887",
7     "email": "joao@alura.com.br"
8   }
9 ]

```

1. []

2. "usuario": {

3. "id": 1,

4. "nome": "João da Silva",

5. "dataNascimento": "30/01/1990",

6. "telefone": "11998877887",

7. "email": "joao@alura.com.br"

8. }

9.]

Como a requisição funcionou no Postman, você decide criar a mesma requisição em código C# no projeto da aplicação Xamarin Forms.

Então você começa a implementar a classe `LoginService`, responsável pelo serviço de login e autenticação do usuário:

```

public class LoginService
{
    public async Task<HttpResponseMessage> FazerLogin(Login login)
    {
        using (var cliente = new HttpClient())
        {
            cliente.BaseAddress = new Uri("https://aluracar.herokuapp.com");

            HttpContent camposFormulario = ???

            return await cliente.PostAsync("/login", camposFormulario);
        }
    }
}

```

Porém, ainda falta preencher corretamente os campos para autenticação (email e senha) do usuário. Complete o código acima para que ele funcione corretamente, de acordo com a especificação da AluraCar .

